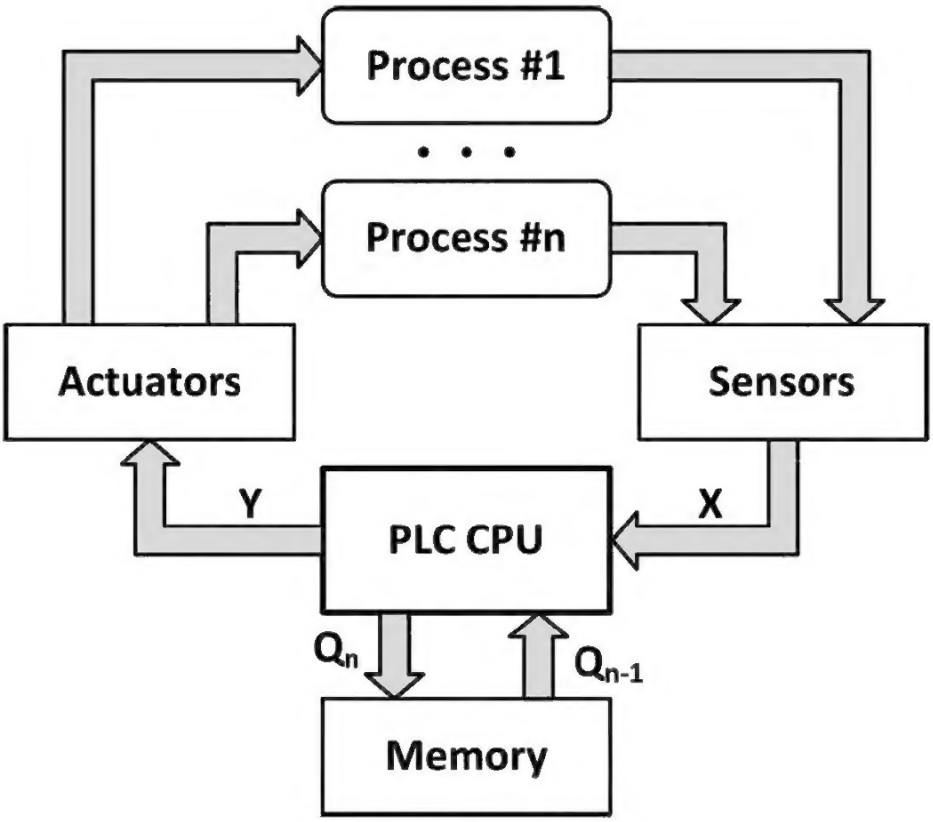
PLC system process



Index:

- Overall System Development Method
- PLC Programming Methods (Languages)
- What Is a Logic Gates ?
- Non-IC Logic Gates Methods
- PLC arduino
- Dedicated Hardware Solutions
- Interfacing Separate Units
- Arduino Software Tools

The process of making a PLC system involves a structured engineering lifecycle, from initial design to commissioning.

- Overall System Development Method:

The general methodology for creating a PLC system involves the following steps:

1. Identify Project Requirements:

Define clearly what the system needs to do, including control tasks, technical conditions, safety requirements, and constraints.

2. Determine I/O Devices and Signals:

Identify all field input devices (sensors, buttons, limit switches) and output devices (motors, valves, lights). Classify them as digital or analog and count the required I/O points.

3. Select PLC Hardware:

Choose an appropriate PLC model, CPU type, memory capacity, I/O modules, and any necessary communication interfaces based on the system requirements.

4. Design Electrical Schematics & Panel Layout:

Create detailed electrical drawings showing component placement, power distribution, and wiring diagrams that adhere to safety standards.

5. Install and Wire Hardware:
Mount all components (PLC, terminal blocks, circuit breakers) in the control panel and wire them according to the schematics. Perform initial power checks for shorts.
6. Develop Control Program:
Design the application software (the control logic) using appropriate programming methods and languages.
7. Test and Debug:
Simulate the program using software tools, then conduct thorough testing with the actual hardware (initial testing, quality assurance, safety function checks) to ensure it operates as intended.
8. Finalize and Document:
Lock the program to prevent unauthorized changes, prepare detailed technical documents, and provide training for personnel.
9. Commissioning and Operation:
Conduct on-site debugging and pre-production tests, and eventually hand over the project for full operation.

- PLC Programming Methods (Languages):

https://archive.org/details/a-programmable-logic-controller-plc

- What Is a Logic Gates ?

https://archive.org/details/what-is-logic-gates

- Non-IC Logic Gates Methods:

https://archive.org/details/non-ic-logic-gates-methods

- PLC arduino:

The integration of PLCs and Arduino can be achieved using specific hardware that combines both technologies (such as the Arduino Opta or Industrial Shields products) or by connecting separate units using standard industrial communication protocols like Modbus over RS485 or Ethernet.

- **Dedicated Hardware Solutions:**

1. Arduino Opta:
This is a micro PLC developed by Arduino Pro, combining the robustness of industrial PLCs with the flexibility of the Arduino ecosystem.
It can be programmed using the standard Arduino IDE 2 with C++ or the Arduino PLC IDE which supports all five standard IEC 61131-3 PLC programming languages (Ladder Diagram, Functional Block Diagram, Structured Text, Sequential Function Chart, and Instruction List).
It features robust, secure hardware with communication options including Ethernet, USB-C, RS485, Wi-Fi, and Bluetooth Low Energy, making it suitable for professional industrial automation and IoT applications.
It is available in three variants: Opta Lite (Ethernet, USB-C), Opta RS485 (adds RS485), and Opta WiFi (adds Wi-Fi/Bluetooth).

2. Industrial Shields PLCs:
Industrial Shields manufactures a range of industrial-grade PLCs like the M-Duino and Controllino series that are based on

original Arduino boards (like the Arduino Leonardo or Mega).
These products feature up to 20 I/Os and support industrial protocols such as RS232, RS485, I2C, SPI, and Modbus.
They are designed for professional use, complying with industrial regulations and offering better durability and maintainability compared to standard Arduino boards.
- **Interfacing Separate Units:**
If you want to connect a standard Arduino board to an existing, separate PLC, you can use the following methods:
* **Communication Protocols:**
Utilize industrial communication protocols like Modbus (RS485 or TCP/IP via Ethernet).
Arduino libraries are available (e.g., Settimino library for Siemens PLCs) to facilitate this.
Other protocols like CAN communication or Profibus-DP can also be used with appropriate shields or adapters.
* **Direct I/O Interfacing:**
You can directly connect the digital or analog I/O pins of the Arduino and PLC.

Voltage compatibility is crucial; you will likely need voltage dividers or optoisolators to protect the Arduino's delicate 5V (or 3.3V) input pins from higher PLC voltages (typically 24V).

- **Arduino Software Tools:**

1. Arduino PLC IDE:
This dedicated software allows you to program compatible Arduino PLCs (Opta, Portenta Machine Control) using standard IEC 61131-3 languages. A lifetime license key is required for some devices/ features.

2. OpenPLC:
This open-source software platform allows you to turn a standard Arduino board (like the Arduino Uno) into a low-cost PLC that can be programmed with ladder logic and other IEC 61131-3 languages.

3. Arduino Cloud:
Compatible PLC hardware can integrate seamlessly with the Arduino Cloud for IoT projects, enabling remote monitoring and control functionalities.

By: Haytham Zeidan

https://archive.org/details/@wazefapress

Resources:

https://www.arduino.cc/education/plc-starter-kit

https://www.digikey.lv/en/maker/blogs/2023/arduino-plc-ide-software-overview

https://www.arduino.cc/pro/software-plc-ide

https://www.industrialshields.com/industrial-plc-based-on-arduino-original-boards-automation-solutions-20ios

https://www.arduino.cc/pro/hardware-arduino-opta/

https://www.instructables.com/INTEGRATING-ARDUINO-INTO-PLC-SYSTEM/

https://forum.arduino.cc/t/interfacing-plc-to-an-arduino/210081

https://www.industrialshields.com/blog/arduino-industrial-1/how-to-connect-arduino-based-plc-with-siemens-s7-plc-family-equipped-through-ethernet-60

https://docs.arduino.cc/software/plc-ide/tutorials/plc-ide-cloud-support/

https://www.plcgurus.net/mastering-automation-with-arduino-plc-ide/

https://control.com/technical-articles/plc-ladder-logic-on-an-arduino-introduction-to-openplc/